



# **ASHESI UNIVERSITY COLLEGE**

## **E-COMMERCE FOR CANTEEN SERVICES**

### **APPLIED PROJECT**

B.Sc. Management Information Systems

**Sylvia Monalisa Engmann**

**2017**

**ASHESI UNIVERSITY COLLEGE**

**E-Commerce for Canteen Services**

**APPLIED PROJECT**

Applied Project submitted to the Department of Computer Science, Ashesi  
University College in partial fulfilment of the requirements for the award of  
Bachelor of Science degree in Management Information Systems

**Sylvia Monalisa Engmann**

**April 2017**

## DECLARATION

I hereby declare that this Applied Project is the result of my own original work and that no part of it has been presented for another degree in this university or elsewhere.

Candidate's Signature:

.....

Candidate's Name:

.....

Date: .....

I hereby declare that preparation and presentation of this Applied Project were supervised in accordance with the guidelines on supervision of Applied Project laid down by Ashesi University College.

Supervisor's Signature:

.....

Supervisor's Name:

.....

Date: .....

## **Acknowledgements**

To my **God**, for granting me grace and strength these past four years, and to do this project. I couldn't have gotten this far without you.

To my family, most especially my mother who was (and still is) a constant encouragement, whenever I picked up the phone to complain.

To my supervisor, Mr Osafo-Mafo, thank you for pushing me on, for all the advice and for making you sit down hours on end. Thank you.

To my friends, who have been there for me though all the times and seasons, I'm grateful.

## **Abstract**

Food ordering applications for restaurants and eateries in Ghana have features such as make orders, select options for delivery or pickup, and provide various options of payment. The context of this project is to ease the pressure on Ashesi University College's canteen servers, give customers flexibility in buying food and take into consideration customizations tailored for Ashesi community (such as the meal plan system).

Ashesi has limited eating services, and with a growing population, using the cafeteria services at certain times of the day (where classes are over during lunch times) is a problem. The community will be able to view and order meals, select a delivery or pickup option, and make payments via delivery or otherwise (with the meal plan). There is also a review and rating system, to give instant feedback to the cafeteria services on the meals, and allow the customers to make an informed choice on what they want to eat.

## Table of Contents

DECLARATION .....	i
Acknowledgements.....	ii
Abstract .....	iii
Table of Contents .....	iv
Chapter 1: Introduction .....	1
1.1 Introduction.....	1
1.2 Motivation.....	1
1.3 Problem Statement .....	2
1.4 Proposed Project .....	2
1.5 Benefits .....	2
1.6 Project Objectives .....	3
1.7 Background and Related Works .....	3
1.7.1 Automated Food Ordering System with Real-Time Customer Feedback. ....	4
1.7.2 The Application of Wireless Food Ordering System.....	4
Chapter 2: Requirements .....	5
2.1 Project Scope .....	5
2.2 Requirements .....	5
2.3 User Requirements.....	5
2.4 System Requirements .....	6
2.4.1 Functional Requirements .....	6
2.4.2 Non-functional Requirements .....	7
2.4.3 Summary of Requirements .....	7
Chapter 3: Architecture and Design.....	8
3.1 The Presentation Layer .....	8
3.2 The Web Server Layer .....	8
3.3 Database Layer .....	9
3.3.1 Database Design .....	9
Chapter 4: Implementation .....	11
4.1 Tools .....	11
4.1.1 The Database.....	11
4.1.1.1 MySQL .....	11

4.1.2 The Application Server .....	12
4.1.2.1 PHP .....	12
4.1.3 Client/Application.....	12
4.1.3.1 HTML .....	12
4.1.3.2 CSS .....	12
4.1.3.3 JavaScript.....	13
4.1.3.4 GoNative.io .....	14
4.1.3.5 Firebase Cloud Messaging System .....	14
Chapter 5: Testing and Results .....	17
5.1 Development Testing .....	17
5.1.1 Unit (Functional) Testing.....	17
5.2 Compatibility Testing .....	20
5.3 Challenges.....	20
Chapter 6: Conclusions and Recommendations .....	21
6.1 Conclusions.....	21
6.2 Recommendations and Future Work .....	21
References.....	22

# **Chapter 1: Introduction**

## **1.1 Introduction**

The advent of technology has brought about many different changes to the manual and usual system of doing things in many areas of life. One area that has been hugely affected is the traditional system of food ordering, in restaurants and fast food places, which has also impacted delivery services.

Technology has introduced a system where with the use of web applications or more commonly, mobile applications, users can make orders for food online, make payments, and even check the status of their orders without having to make a phone call, or going to the restaurant. Web and mobile food ordering applications have come to stay because of the convenience it offers the users. And for those who still prefer the traditional means, there are options to order the food, let the restaurant know what time the customer would be coming in (which may be more of a booking).

## **1.2 Motivation**

The potential of web and mobile food ordering applications to Ashesi is evident in the long queues during peak hours of its cafeterias. The numbers at Ashesi are increasing, yet we have the service of only two food cafeterias, and a snack bar. In 2013, 171 out of 1,427 applicants were admitted to the class of 2017(Ashesi University College, 2013). In 2014, 171 out of 880 applicants were admitted (Ashesi University College, 2014). In 2016, the numbers went up to 278 out of 1,217 applicants (Ashesi University College, 2016). The increase in numbers, coupled with the inadequate infrastructure to seat large numbers, is a contributing factor to the queues at the cafeterias during peak hours.



### **1.3 Problem Statement**

An observation at the cafeterias at some points in a day reveals that several people cannot be comfortably seated at the tables provided, due to limited space and infrastructure. These periods in the day are ‘peak times’, and a study done in the 2015-2016 academic year by Dorothy Sarpong-Kumankoma on *Exploring the Use of Predictive Analytics in Reducing Customer Wait Time in a College Cafeteria* revealed that there was an average of 22.4 people at the cafeteria every thirty minutes in these ‘peak times’. An average common peak period determined by the highest number of people at the cafeteria was between 12:45pm and 2:45pm (Sarpong-Kumankoma, 2016).

Patrons of the cafeteria usually wish to know what food is available before making their way there, as well as knowing if their desired meal is good (or otherwise) before they buy. There’s also the need to reduce the incidence of queues in the cafeteria.

### **1.4 Proposed Project**

The proposed project is a hybrid application (a web application that is converted into a mobile application) that will allow the Ashesi Community to order food online from any of the cafeterias, make payments, and either pick up their order or have it delivered based on the user preference.

### **1.5 Benefits**

The system is expected to ease pressure at the cafeterias during peak hours, give the user the opportunity to see what meals are available, place orders and skip the process of having to join a queue for food.

## **1.6 Project Objectives**

The project seeks to design and implement a system that can be used by the Ashesi community to order food from the cafeterias. In addition, some of the non-technological impacts that would be looked at would be how online food ordering will impact customers, and whether there's an effect on customer satisfaction, and how the online ordering will impact the business, as well as what adjustments will have to be made to the operational structure of the business.

## **1.7 Background and Related Works**

As far as our research, online food ordering system for school canteen services in Ghana is not usual. Most cases reviewed were colleges in the US, and these made use of an online cafeteria menu viewing system in an application with other functionalities for the school. A case study example is the Azusa Pacific University, which has three different applications: APU Events, Mobile Trolley and APU Mobile, and the latter contains Dining Menus, Campus Location and Calendars, recent news etc. (Azusa Pacific University, n.d.). The APU Mobile application shows students the menu that is being served in the school cafeteria.

There however, exists a system in High Schools (and again this is more prominent in the US) where parents have the opportunity order in advance and pay for lunch for their children, for a specified period (a month, a term or a whole school year). It allows them to be able to choose menus based on their children's eating habits, allergies etc. A case example is MySchoolAccount which has a meal pre-order system which allows a school to place their school lunch menu up and the parents can then order their ward's lunch in advance, and payment can be made upfront as well (MySchoolAccount, n.d.). The motivation for this being done in High Schools in the US was to curb the incidence of theft and to have a more cashless system.

Here in Ghana, in Senior Secondary Schools that have a boarding school system, food fees are paid upfront at the beginning of every school term and students then go to the dining hall to eat every day, which works more like Ashesi's meal plan system.

#### **1.7.1 Automated Food Ordering System with Real-Time Customer Feedback.**

The authors of the research paper sought to automate the food ordering process in restaurants and integrate real time customer feedback (AOS-RTF). This is done with wireless data access to servers. In this system, the users will have menu details. The order details are then sent via wireless transmission to a central database which will be sent to the kitchen and cashier. The restaurant owner can manage and effect menu changes at any point in time and this again will be synced to the application for the user. The wireless application is a means of convenience, improving efficiency while saving time, reducing human errors and real time customer feedback (Tanpure, S. S., Shidankar, P. R., & Joshi, M. M., 2013).

#### **1.7.2 The Application of Wireless Food Ordering System.**

This system was designed by the owners of the research to check for the feasibility of a wireless food ordering system, which will allow users make food orders online, and have it delivered to them. The outcome of the research showed that it was feasible (Khairunnisa K., Ayob J., Mohd. H. A., Wahab M., Ayob E., Ayob M. I., & Ayob M. A., 2009).

## **Chapter 2: Requirements**

### **2.1 Project Scope**

The system has two main components: a server side which will process the information and allow for the connection to the database and the user/client side: the Ashesi community users, and the kitchen (cafeteria) users. The focus of this project is the Ashesi community users.

### **2.2 Requirements**

Requirements were first gathered from students, to determine how a food ordering application can be customized for the community. The other point of requirements gathering was from existing food ordering applications like Jumia Foods, and (because it is an ecommerce application) other e-commerce web applications (on a mobile device) e.g.

‘walmart.com’ as well as ‘urbanmakes.com’.

### **2.3 User Requirements**

These requirements are what the Ashesi Community (henceforth referred to as the user(s)) would expect to see and make use of, in the usage of the application.

- The application should allow users to see the meals that are available at the cafeteria at every point of the day.
- The application should allow users to place an order for any meal available on the menu, and receive a confirmation (or otherwise) of their order.
- The application should give the users options in receiving their order, and payment options as well.

- The application will allow users to give and read reviews on meals that have been bought during the day.

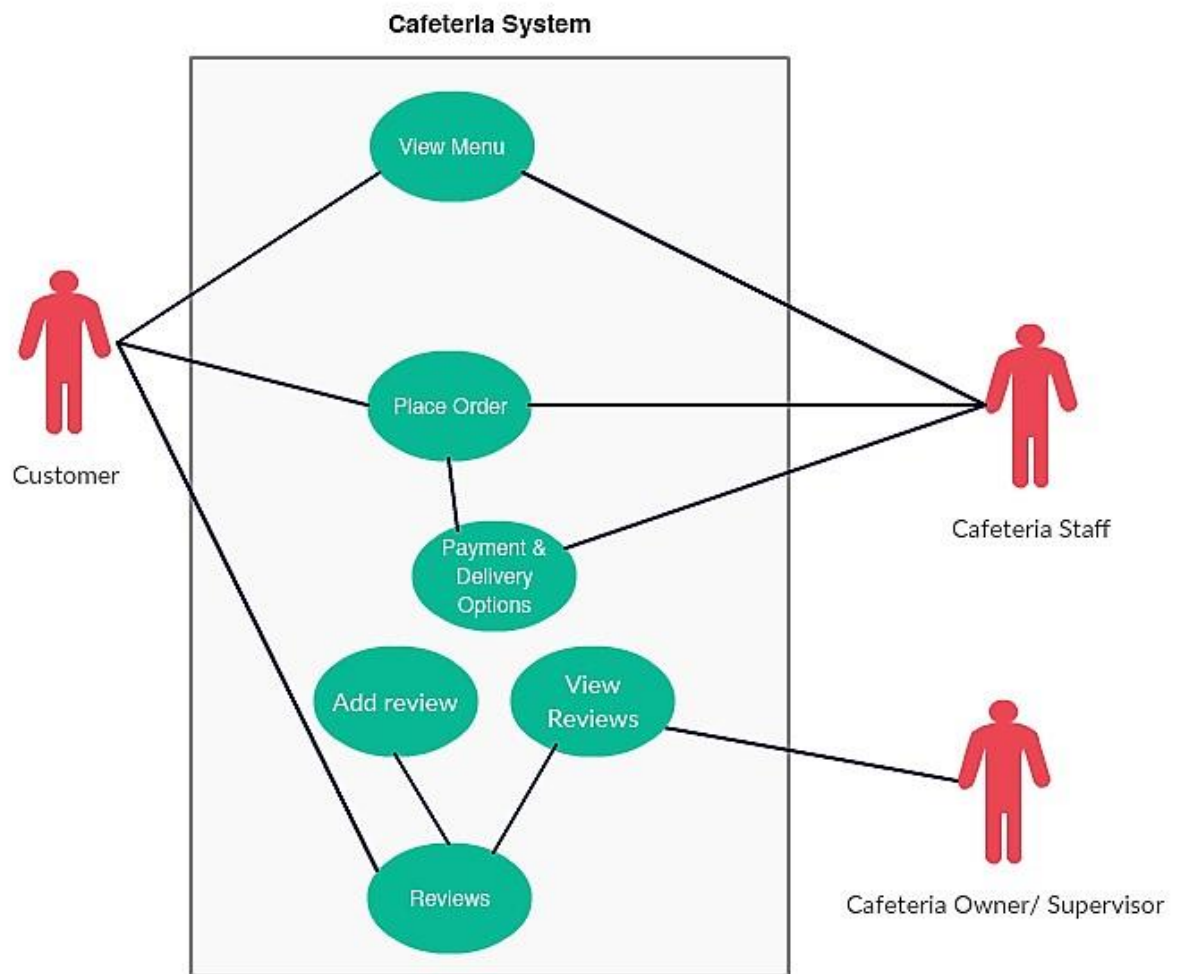


Figure 2.1: Use case diagram for system (User requirements)

## 2.4 System Requirements

### 2.4.1 Functional Requirements

- The application should have a sign-up system for new users.
- The application should have a login system to enable the users access the menu available, and is better suited to help the user keep track of the meals ordered.

- The application should allow users to place orders, and be able to view orders placed, and the status of the order (confirmed or otherwise, and if confirmed, when order will be ready for pick up or delivery).
- The application should have a real-time system for sending and displaying reviews.

#### **2.4.2 Non-functional Requirements**

- Usability – The application should be easy to use and navigate
- Performance (High Response time): In sending users notifications when orders are confirmed, and otherwise.

#### **2.4.3 Summary of Requirements**

The application should allow customers to view the menu available for the various cafeterias, place orders and either pick it up or have it delivered. They should also be able to write and read reviews on any meal available.

## Chapter 3: Architecture and Design

System design and architecture is the process of defining components, modules, interfaces and data for a system to satisfy specified requirements (The MITRE Corporation, n.d.). Software Architecture is the blueprint that developers use in building a system, and clearly defines the different parts of the system and how they interact.

The application would be based on the client/server model architecture, the client side being a web application converted to a cross-platform mobile application, and the server made up of the network server and the database server.

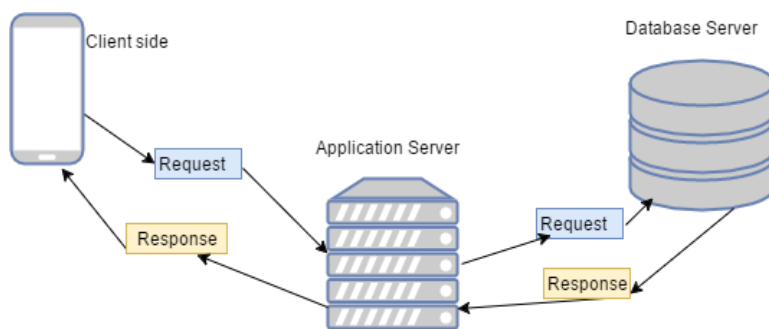


Figure 3.1: High Level Architecture

### 3.1 The Presentation Layer

The client side is the application with which the client will communicate with the server and make orders, receive confirmation (or otherwise) of the order, and receive notification of order readiness.

### 3.2 The Web Server Layer

The application server contains the means of communication between the database server and the client application. It relays the requests made by the client application to the

database server, (and to the cafeteria side application) and then relays the responses from the database server back to the client.

### 3.3 Database Layer

The database server is responsible for all the tasks pertaining to the data storage, manipulation, analysis etc.

#### 3.3.1 Database Design

The database design of the user/client application is mainly based on the users, the orders they make and the reviews given. Their ability to choose their preferred cafeteria, as well as view the menu (meals) available will come from the database of the cafeteria (which is being implemented by a second person).

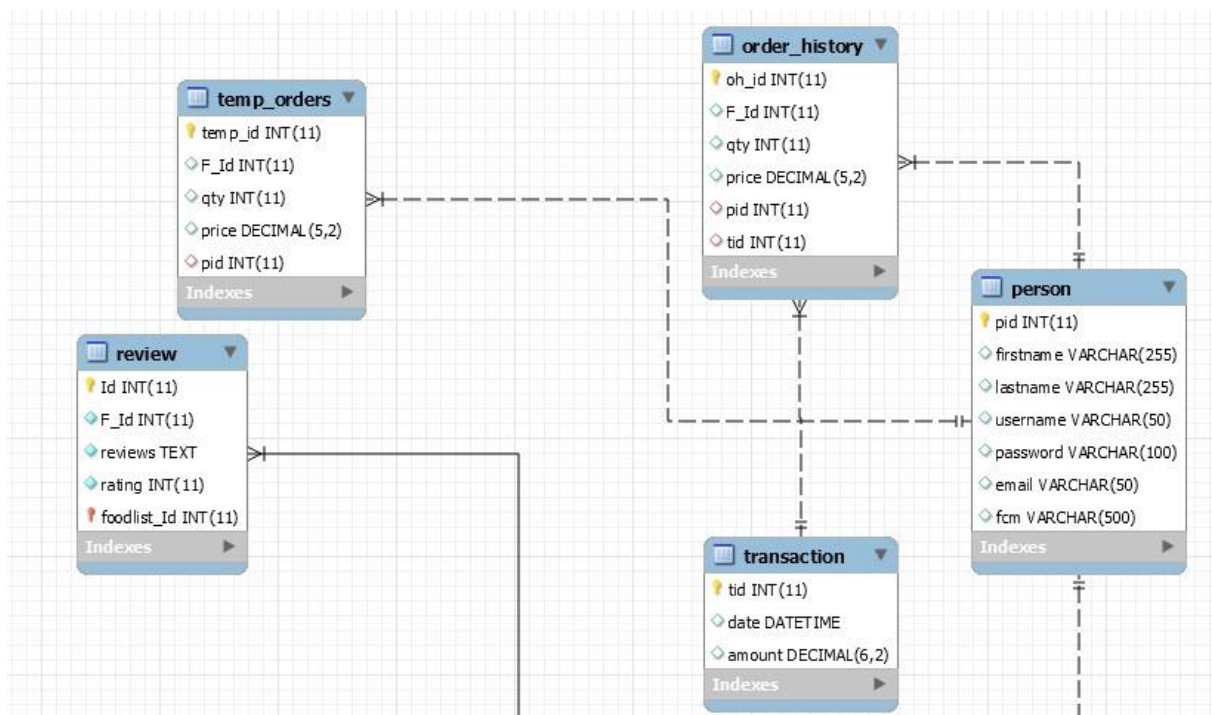




Figure 3.2: Database design (ER diagram)

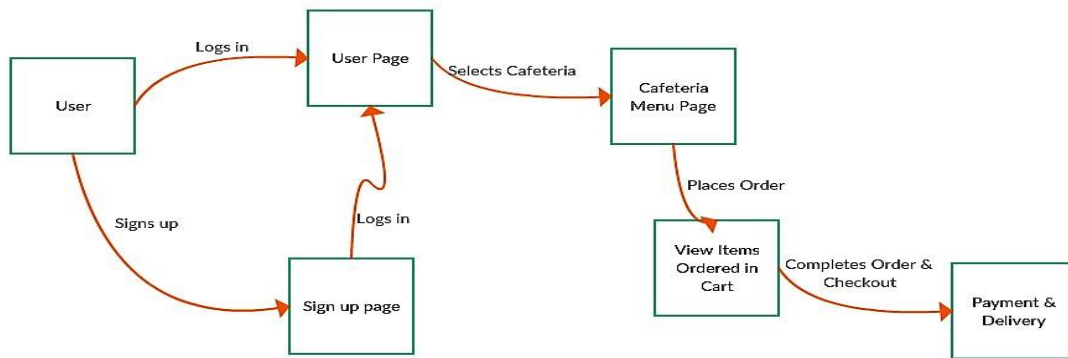


Figure 3.3: High-level user interface design

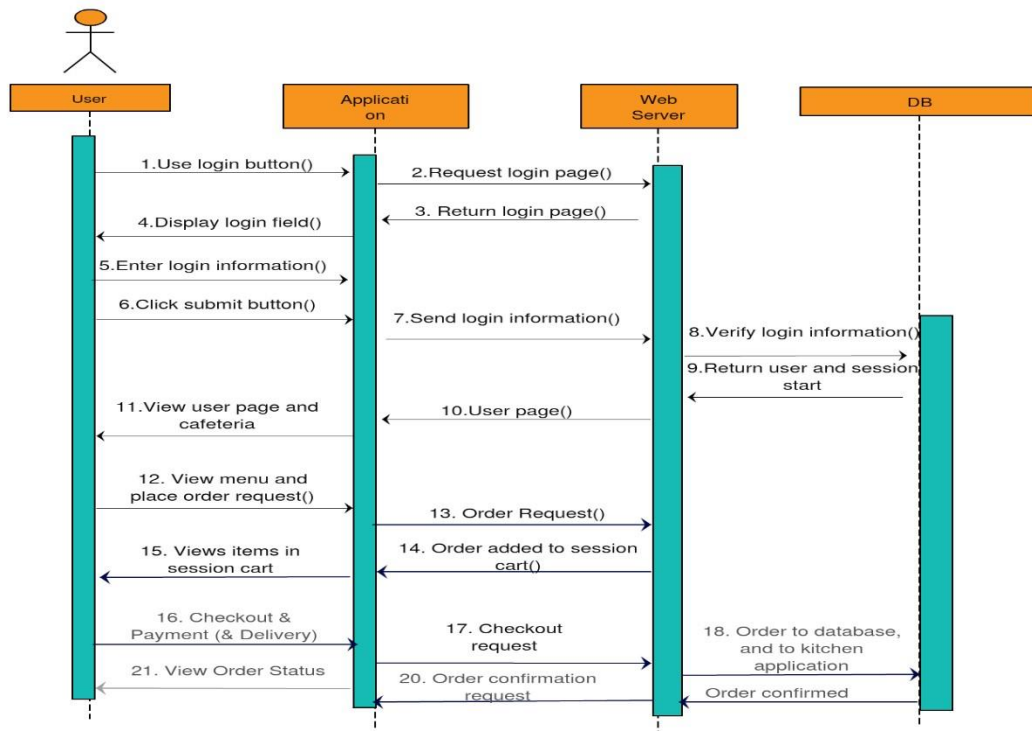


Figure 3.4: Sequence diagram

## Chapter 4: Implementation

The implementation of the application was done using the following tools and libraries:

### 4.1 Tools

#### 4.1.1 The Database

##### 4.1.1.1 MySQL

MySQL is an open source relational database management system that uses Structured Query Language (SQL) to process (define and manipulate) data in a database. MySQL is written in C and C++ and is compatible with all the major operating systems, i.e. Windows, MAC OS and Linux systems (Techopedia, n.d.).

MySQL can be used with a lot of different applications, but is mostly associated with Web applications and is a component of an open source enterprise stack called LAMP. This consists of **L**inux as the operating system, **A**pache as the web server, **M**ySQL as the relational database management system and **P**HP as the scripting language (TechTarget, n.d.). These are the components that will be used in the development of this application as well, since its primarily a web application (except the operating system).

The tables used in this project are: **Person** table, **Temp\_Order** table, **Order\_History**, **Transaction** table and **Reviews** table.

- **Person** table: Contains the person name, the preferred username and password (that will be used for sign in) and email address.
- **Temp\_Order** table: Contains the meal chosen, and the quantity bought, before the customer confirms and checks out. This is also the table from which details will be sent to the cafeteria for confirmation

- Order\_History table: Contains the orders made, and the quantity bought, after the customer's order has been confirmed. The customer can view all their order history on this page.
- Reviews Table: Contains the reviews made about a meal, on a date.
- Transaction table: Contains the date on which a transaction was made, and the total amount.

### **4.1.2 The Application Server**

#### **4.1.2.1 PHP**

Hypertext PreProcessor (PHP), is an open source server side scripting language usually used in web development and can be embedded into HTML (which would also be used, and explained) (PHP, n.d.). It also interacts with many different database languages (including MySQL) so it's a preferred choice for server-side scripting, and is compatible with Apache servers and runs on Windows, Linux and Unix servers (Blue Line Media, 2017).

### **4.1.3 Client/Application**

#### **4.1.3.1 HTML**

The HyperText Markup Language (HTML), describes and defines static content on a webpage (Mozilla Developer Network, n.d.). HTML in the project was used to define all the various (static content) part of the application.

#### **4.1.3.2 CSS**

Cascading Style Sheets are a stylesheet language used to describe the actual style of the HTML defined on the webpages. In this project, I made use of the Materialize CSS

Framework (which is based on Google's Material design), which has its own stylesheet, as well as JavaScript files (which I'll speak of next) which aid in the dynamic nature of the HTML and CSS layouts.

```
<link rel="stylesheet" href="css/materialize.css">
<link rel="stylesheet" href="css/material-design-iconic-font.min.css">
<link rel="stylesheet" href="css/style.css">
<link rel="stylesheet" href="css/jquery.mobile-1.4.5.css">
<link rel="stylesheet" href="css/jquery.mobile.structure-1.4.5.css">
<link rel="stylesheet" href="css/jquery.mobile.theme-1.4.5.css">
<link href="https://fonts.googleapis.com/icon?family=Material+Icons" rel="stylesheet">
```

Figure 4.1: CSS setup

#### 4.1.3.3 JavaScript

JavaScript is an object-oriented programming language that when included in an HTML document, provides dynamic interactivity on webpages (Mozilla Developer Network, n.d.). It runs on the client side of the application and doesn't need to make requests to the server. This project makes use of jQuery's ajax function, as well as Materialize JavaScript function for the side navigation bar. Below is the setup (as well as some examples) of the JavaScript files.

```
<script src="scripts/jquery.js"></script>
<script src="scripts/materialize.min.js"></script>
<script src="scripts/jquery.mobile-1.4.5.min.js"></script>
<script type="text/javascript" src="scripts/platformOverrides.js"></script>
```

Figure 4.2: JavaScript setup

```
<script>
(function($) {
  $(function() {
    $('.button-collapse').sideNav();
  }); // end of document ready
})(jQuery); // end of jQuery name space
</script>
```

Figure 4.3 JavaScript function (1)

```
var url = "functions.php?cmd=1&fname="+fname+"&ln  
$.ajax(url,  
  {async:true,complete:signupComplete}  
  );
```

Figure 4.4 JavaScript function (2)

#### 4.1.3.4 GoNative.io

Another framework that was used, is the GoNative.io Framework (that will be used in converting the web application into a cross-platform mobile application). The technology behind GoNative is built to take any existing web application and seamlessly convert it into native apps for Android and iOS.

#### 4.1.3.5 Firebase Cloud Messaging System

To allow for easy communication between the customer's app and the kitchen application, the Firebase Cloud Messaging System was used. Firebase Cloud Messaging (FCM) is a cross-platform messaging solution that delivers messages at no cost. It's available for iOS, Android as well as Web applications (Google, 2017).

FCM needs authorization to send messages to an application, and it uses an identification number (shown as `gcm_sender_id` in the image below) to establish that it can send messages to the application.

```

{
  "short_name": "Bon Appetit",
  "name": "Bon Appetit",
  "icons": [
    {
      "src": "png/app/assets/mobileweb/appicon.png",
      "type": "image/png",
      "sizes": "128x128"
    }
  ],
  "start_url": "index.php",
  "///": "Some browsers will use this to enable push notifications.",
  "///": "It is the same for all projects, this is not your project's sender ID",
  "gcm_sender_id": "103953800507"
}

```

Figure 4.5: Firebase Cloud Messaging (authorization method)

The method below requests permission from the user to receive notifications. If this is successful, the user is assigned a token, which will be stored in the database. This token will be used as the receiving point for all notifications to the user. Each user will have a unique token.

```

messaging.requestPermission()
.then(function() {
  console.log('Notification permission granted.');
```

---

```

  // TODO(developer): Retrieve an Instance ID token for use with FCM.
  // ...
})
.catch(function(err) {
  console.log('Unable to get permission to notify.', err);
});

```

Figure 4.6: Firebase Cloud Messaging (Method for permission)

```

// Get Instance ID token. Initially this makes a network call, once retrieved
// subsequent calls to getToken will return from cache.
messaging.getToken()
.then(function(currentToken) {
  if (currentToken) {
    sendTokenToServer(currentToken);
    updateUIForPushEnabled(currentToken);
  } else {
    // Show permission request.
    console.log('No Instance ID token available. Request permission to generate one.');
```

---

```

    // Show permission UI.
    updateUIForPushPermissionRequired();
    setTokenSentToServer(false);
  }
})
.catch(function(err) {
  console.log('An error occurred while retrieving token. ', err);
  showToken('Error retrieving Instance ID token. ', err);
  setTokenSentToServer(false);
});

```

Figure 4.7: Firebase Cloud Messaging (Method for assigning token to user)



Figure 4.8: Homepage

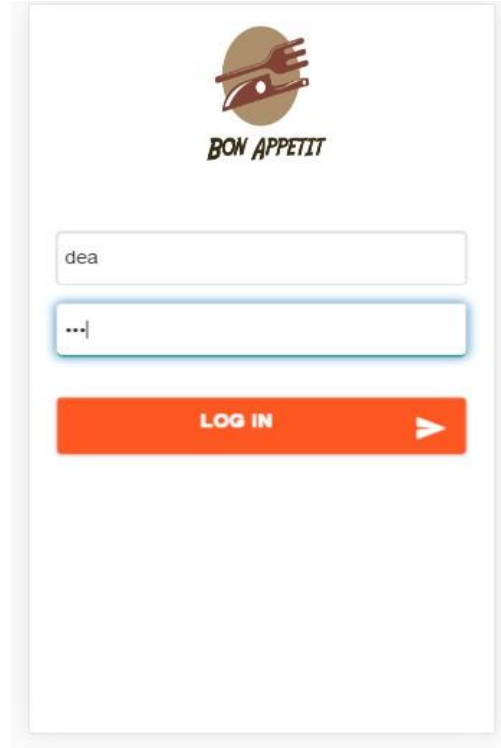


Figure 4.9: Login page

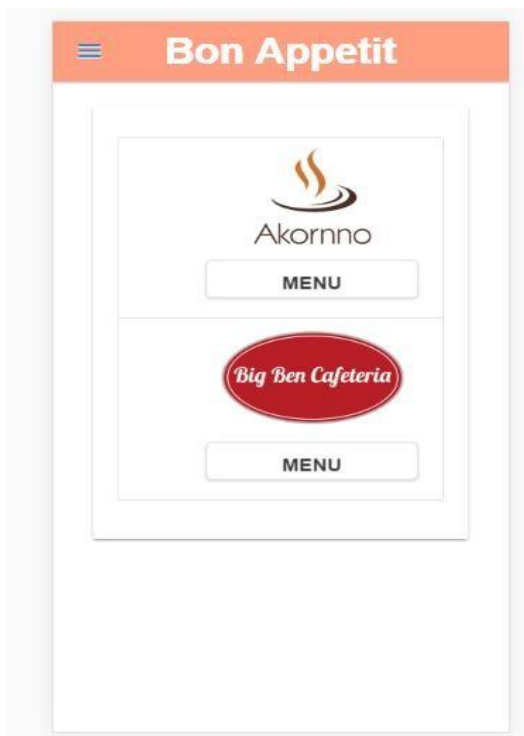


Figure 4.10: Cafeteria

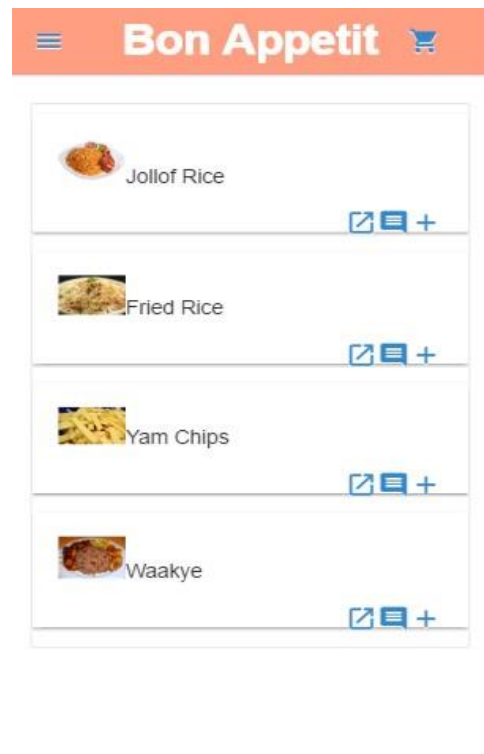


Figure 4.11: Menu

## Chapter 5: Testing and Results

### 5.1 Development Testing

Development testing refers to all tests carried out during the development stage of the software, and this usually done by the programmer. Below are the tests that were carried out during the development stage.

#### 5.1.1 Unit (Functional) Testing

The parts of the software that were tested were registering a new user, log in an existing user placing an order (into the cart) and giving reviews. For example, in the login system, when a user logs in, the username stored in a session is displayed in the side navigation bar. Below are screenshots of the successful login test.

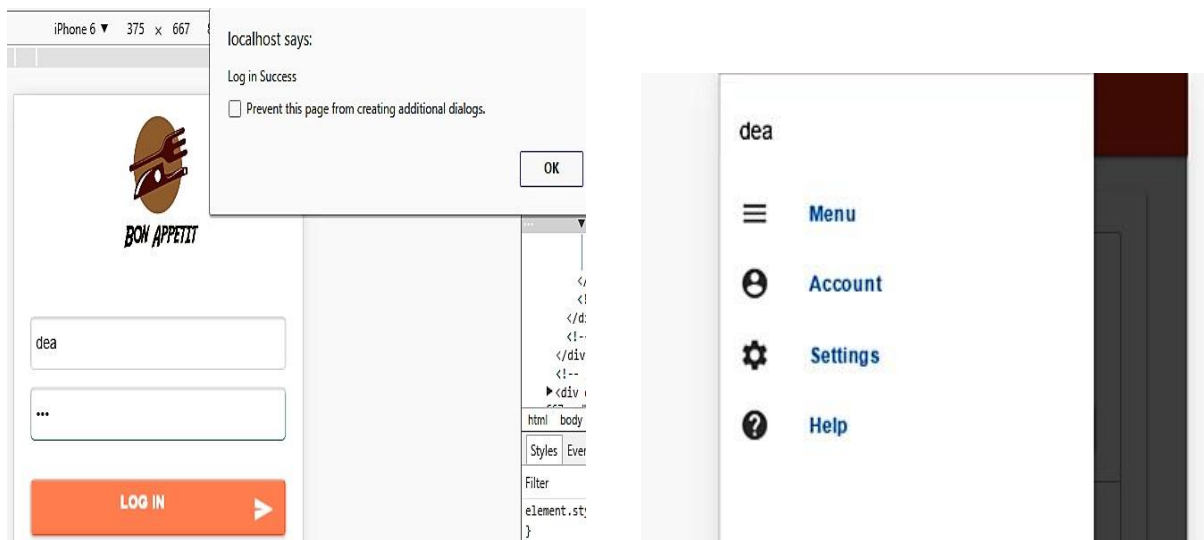


Figure 5.1: Login test

The user can add a menu to the cart, edit the items in the cart before finally completing the order and checking out to place the order. The images below show the user being able to



choose a food item, specify the type (for a full or half-portioned meal), specifying a quantity  
place a food item in the cart (edit or remove as needed), and finally checkout.

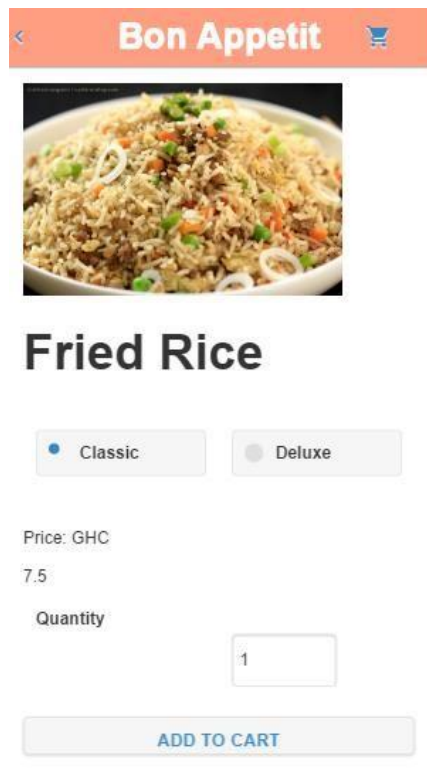


Figure 5.2: Meal details



Figure 5.3: Cart

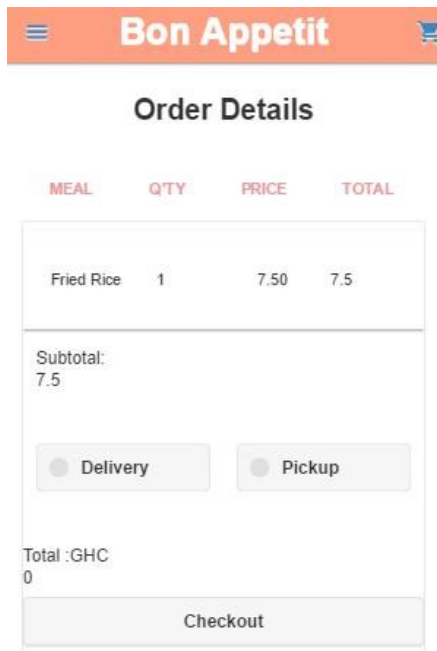


Figure 5.3: Pre-checkout

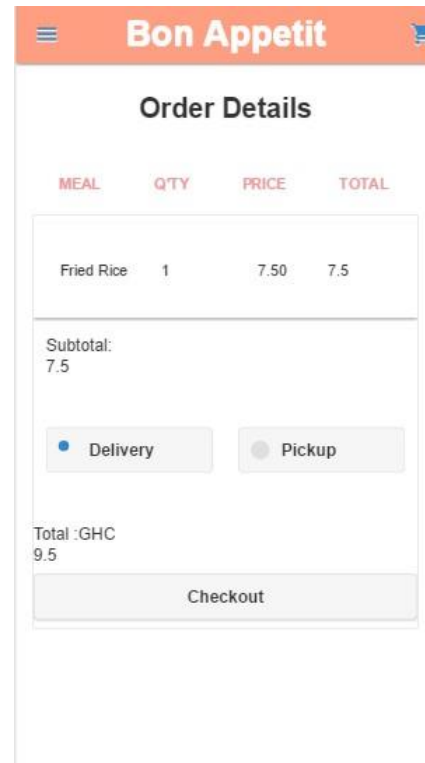


Figure 5.4: Post-checkout

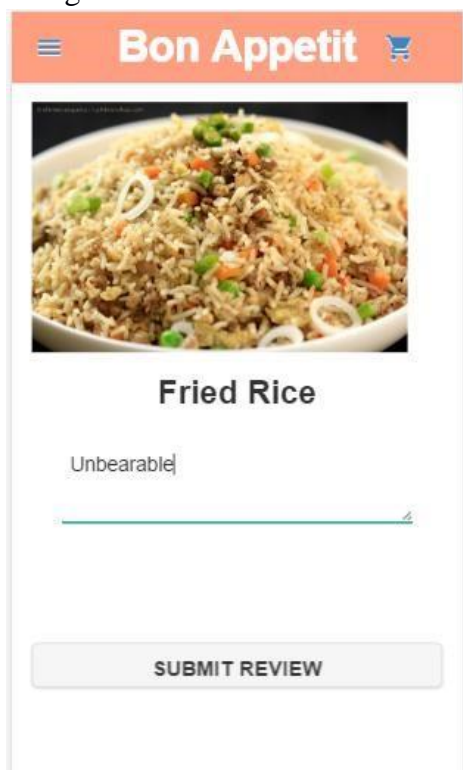


Figure 5.5: Pre-review

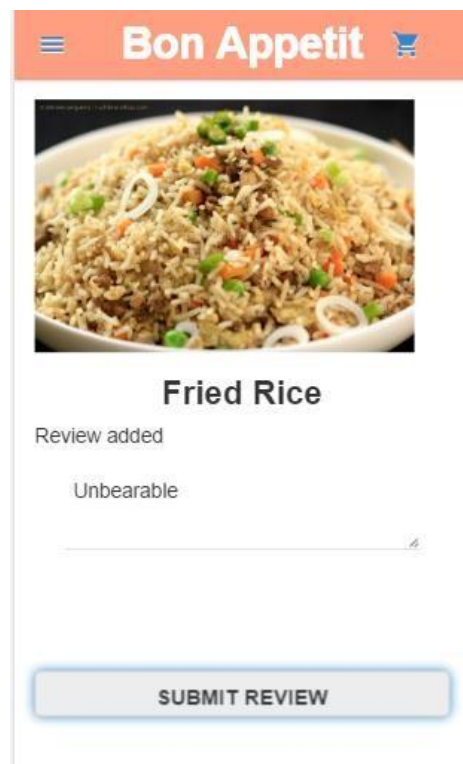


Figure 5.6: Post-review

## 5.2 Compatibility Testing

This test was carried out to check the appearances of the interfaces on various mobile platforms, via the Google Chrome **inspect feature**. It was also tested on the Firefox browser.

## 5.3 Challenges

The biggest challenge I faced was implementing the shopping cart. I wanted to avoid using an existing cart, so I set out to design the service from scratch, and it took most of my time. Another challenge was setting up the Firebase Cloud Messaging system, to allow a notification to be sent when a customer places an order, as well as receiving a notification when an order is ready.

## **Chapter 6: Conclusions and Recommendations**

### **6.1 Conclusions**

Even though the application was designed with the Ashesi community at hand, it can be used anywhere because of the flexibility of being able to add a cafeteria, to add and remove meals, as well as login and register to place orders.

One limitation is the implementation of the payment system. The user, must make payment while checking out. If the order is not confirmed or approved, the student will have to be reimbursed. Another limitation is the reliance on the internet, to allow for the notification system to be effective, and in total the whole application to be fully functional.

### **6.2 Recommendations and Future Work**

The payment system can be improved upon to include a feasible system for non-meal plan orders, with the use of external payment systems like MPower or SlydePay. There can also be improvement in the user interfaces, to include more images. The application for now only works with cash-on-delivery or cash-on-pickup payment systems, so there would have to be integration of the meal plan system, in addition to the external payment system mentioned early on. Due to limitations of staff number at the cafeterias, there also needs to be an external (non-cafeteria staff) delivery system in place to be able to deliver orders.

## References

- Ashesi University College. (2013). Ashesi's admissions acceptance rate drops to 12%, applications up by 74% - Homepage - Ashesi University College. Retrieved from <http://www.ashesi.edu.gh/news-and-events/news/1716-ashesi-s-admissionsacceptance-rate-drops-to-12.html>
- Ashesi University College. (2014). The anatomy of Ashesi's Class of 2018 - Homepage - Ashesi University College. Retrieved from <http://www.ashesi.edu.gh/stories-andevents/1880-the-anatomy-of-ashesi-s-class-of-2018.html>
- Ashesi University College. (2016). The anatomy of Ashesi's Class of 2020 - Homepage - Ashesi University College. Retrieved from <http://www.ashesi.edu.gh/stories-andevents/2306-ashesi-class-of-2020-anatomy.html>
- Azusa Pacific University. (n.d.). APU Mobile Apps - Azusa Pacific University. Retrieved from <http://www.apu.edu/mobile/>
- Blue Line Media. (2017). Why do we use PHP? Retrieved from <https://www.bluelinemedia.co.uk/blog/entry/web-design/blog/why-php>
- Google. (2017). Set Up a JavaScript Firebase Cloud Messaging Client App | Firebase. Retrieved from <https://firebase.google.com/docs/cloud-messaging/js/client>
- Khairunnisa K., Ayob J., Mohd. H. A., Wahab M., Ayob E., Ayob M. I., & Ayob M. A (2009). The Application of Wireless Food Ordering System. *MASAUM Journal of Computing, Vol 1 Issue 2*.
- Mozilla Developer Network. (n.d.). HTML | MDN. Retrieved from <https://developer.mozilla.org/en-US/docs/Web/HTML>

Mozilla Developer Network. (n.d.). JavaScript basics - Learn web development | MDN.

Retrieved from [https://developer.mozilla.org/en-](https://developer.mozilla.org/en-US/docs/Learn/Getting_started_with_the_web/JavaScript_basics)

[US/docs/Learn/Getting\\_started\\_with\\_the\\_web/JavaScript\\_basics](https://developer.mozilla.org/en-US/docs/Learn/Getting_started_with_the_web/JavaScript_basics)

MySchoolAccount. (n.d.). School payment software for lunch accounts & fee payment.

Retrieved from <https://www.myschoolaccount.com/>

PHP. (n.d.). PHP: What is PHP? - Manual. Retrieved from

<http://php.net/manual/en/introwhatis.php>

Sarpong-Kumankoma, D. (2016). Exploring the use of predictive analytics in reducing customer

wait time in a college cafeteria (Unpublished undergraduate thesis). Ashesi University

College, Accra, Ghana.

The MITRE Corporation. (n.d.). System Design and Development | The MITRE Corporation.

Retrieved from [https://www.mitre.org/publications/systems-engineering-](https://www.mitre.org/publications/systems-engineering-guide/selifecycle-building-blocks/system-design-and-development)

[guide/selifecycle-building-blocks/system-design-and-development](https://www.mitre.org/publications/systems-engineering-guide/selifecycle-building-blocks/system-design-and-development)

Tanpure, S. S., Shidankar, P. R., & Joshi, M. M. (2013). Automated Food Ordering System with

Real-Time Customer Feedback. *International Journal of Advanced Research in*

*Computer Science and Software Engineering*, 1-6.

Techopedia. (n.d.). What is MySQL? - Definition from Techopedia. Retrieved from

<https://www.techopedia.com/definition/3498/mysql>

TechTarget. (n.d.). What is MySQL? - Definition from WhatIs.com. Retrieved from

<http://searchenterpriselinux.techtarget.com/definition/MySQL>